REMARKS

Status of Case

Claims 6-26 are pending in this case.

Interview

Applicant sincerely appreciates the interview on January 22, 2008 for the present application. Applicants present argument consistent with the interview.

Rejection under 35 USC §§ 102, 103

Claims 6-15 and 19 were rejected under 35 U.S.C. § 102(b) as being anticipated by U.S. Patent No. 6,092,194 (Touboul). Claims 16-18 were rejected under 35 U.S.C. § 103(a) as being unpatentable over Touboul in view of U.S. Patent No. 7,124,302 (Ginter et al.).

Applicants present amended claims 6 and 19. Claim 6 recites:

> "a first memory configured to store data and a protection flag, the protection flag indicative of whether the corresponding data requires security protection, wherein the stored data requires security protection and wherein the protection flag indicates that the stored data requires security protection" and
>
> "a determination unit configured to determine, when the application program is executed, whether a target code is included in the restriction code, the target code being an instruction code in the application program to be executed by the execution unit"

See also claim 19 ("storing, in the first memory, data and a protection flag, the protection flag indicative of whether the corresponding data requires security protection, wherein the stored data requires security protection and wherein the protection flag indicates that the stored data requires security protection" and "determining, when the application program is executed, whether a target code is included in the restriction code, the target code being an instruction code in the application program to be executed by the execution unit"). The Touboul reference, either alone or in combination with other cited references, fails to teach or suggest the cited limitations.

A. Different focus of the Touboul reference

As an initial matter, the focus of the Touboul reference is entirely different from that disclosed (and claimed in the present application). The Touboul reference is entirely focused on whether to **download** the content to the internal network (i.e., whether to even store the content

on the device that ultimately uses the content). In the invention as claimed, the content is already stored; rather, the focus is on whether to **execute** the stored content. For example, the Touboul reference teaches the following configuration, as depicted in Figure 1 (reproduced below):

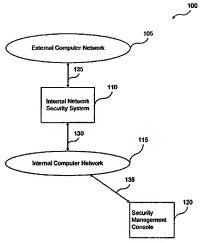


FIG. 1

The Touboul reference teaches that the software to prevent the download is resident on the Internal Network Security System 110. In particular, the Touboul reference teaches that the Internal Network Security System 110 blocks the "downloadable" prior to being stored on the network if it is deemed suspicious. See col. 3, lines 10-13; see also col. 6, line 64 – col. 7, line 2. In this way, the Internal Network Security System 110 can discard the "downloadable", thereby preventing the "downloadable" from even entering the Internal Computer Network 115. Even if one assumes that the Internal Network Security System 110 stores the "downloadable", it is temporary and is not stored on the ultimate device that executes the "downloadable."

B. Recited Limitations not merely rearranging the steps of the Touboul reference

The Office Action states, in the Response to Arguments section, that the invention as claimed is merely a rearranging of the steps disclosed in the the Touboul reference and that the "order of the steps of determining the application is a risk is not a critical limitation and does not provide new or unexpected results." Applicants respectfully disagree. The steps between the Touboul reference and the invention as claimed are significantly different, not merely changing the order of execution. The Touboul reference teaches that the file scheduled for downloaded is scanned prior to storage on the network. If the file is deemed not to be downloaded, it is **never** stored on the internal computer

network. Even if one were to assume that the Internal Network Security System 110 temporarily stores the "downloadable" in order to scan it, that storage is temporary at best and is not even inside the network. In contrast, the invention as claimed does not prevent the storage of the download, but the execution. Specifically, the file according to the present claims is stored on the terminal device. It is only upon execution that the file is scanned to determine whether it contains restricted code. If the file does contain restricted code, the restricted code is not executed. In this way, the claims as currently presented are not merely a reordering of the teachings of the Touboul reference. Rather, the claims as presented are significantly different.

C. Significantly different results from the Touboul reference

The present invention as claimed generates significantly different results than that of the Touboul reference. Two examples of the differences are presented. First, in the Touboul reference, if the file is deemed non-downloadable, it is **never** stored on the terminal device. In contrast, the present claims enable the storing of the file.

Second, if the firewall as taught in Touboul reference deems the file not worthy of download (such as due to a potential virus), the Touboul reference teaches that it is rejected. In contrast, the present invention as claimed still stores the file. If the list of restricted code changes (such as if a particular restricted code present in the stored file is **removed** from the restricted list), after the restricted list is updated, the stored file may be executed. Under the teachings of the Touboul reference, the file would have to be downloaded again. Similarly, if restricted code is **added** to the restricted list, the file would still be checked for execution even if that file may have already been executed.

In this way, the reason behind checking the file for the restricted code at execution becomes clear – the reason is not in order to prevent viruses (such as a firewall in Touboul); rather, the reason is to protect the privacy of the user (such as making sure that certain restricted code, such as accessing private information is not executed). In this way, the invention as claimed allows for the restricted code list to be updated (either to add or delete restricted code).

D. The Touboul reference does not teach determining whether to prevent execution when the application program is executed.

During the interview, the Examiner stated that one may interpret the Touboul reference as teaching that the Internal Network Security System 110, as part of the process of determining if a virus

is present in the "downloadable," executes the "downloadable". Applicants respectfully disagree for several reasons.

First, any reasonable reading of the Touboul reference clearly shows that it does not teach determining whether to prevent execution when the application is executed. The Touboul reference teaches that security program 255 of the Internal Network Security System 110 "controls examination of incoming Downloadables." The following are excerpts from the Touboul reference:

> "The internal network security system 110 examines Downloadables received from external computer network 105, and prevents Downloadables deemed suspicious from reaching the internal computer network 115. It will be further appreciated that a Downloadable is deemed suspicious if it performs or may perform any undesirable operation, or if it threatens or may threaten the integrity of an internal computer network 115 component. It is to be understood that the term 'suspicious' includes hostile, potentially hostile, undesirable, potentially undesirable, etc." Col. 3, lines 10-19.

> "Internal network security system 110 further includes Input/Output (I/O) interfaces 215 (such as a keyboard, mouse and Cathode Ray Tube (CRT) display), a data storage device 230 such as a magnetic disk, and a Random-Access Memory (RAM) 235, each coupled to the signal bus 220. The data storage device 230 stores a security database 240, which includes security information for determining whether a received Downloadable is to be deemed suspicious. The data storage device 230 further stores a users list 260 identifying the users within the internal computer network 115 who may receive Downloadables, and an event log 245 which includes determination results for each Downloadable examined and runtime indications of the internal network security system 110. An operating system 250 controls processing by CPU 205, and is typically stored in data storage device 230 and loaded into RAM 235 (as illustrated) for execution. **A security program 255 controls examination of incoming Downloadables,** and also may be stored in data storage device 230 and loaded into RAM 235 (as illustrated) for execution by CPU 205.

> FIG. 3 is a block diagram illustrating details of the security program 255 and the security database 240. **The security program 255 includes an ID generator 315, a policy finder 317 coupled to the ID generator 315, and a first comparator 320 coupled to the policy finder 317. The first comparator 320 is coupled to a logical engine 333 via four separate paths, namely, via Path 1, via Path 2, via Path 3 and via Path 4. Path 1 includes a direct connection from the first comparator 320 to the logical engine 333. Path 2 includes a code scanner coupled to the first comparator 320, and an Access Control List (ACL) comparator 330 coupling the code scanner 325 to the logical engine 333. Path 3 includes a certificate scanner 340 coupled to the first comparator 320, and a certificate comparator 345 coupling the certificate scanner 340 to the logical engine 333. Path 4 includes a Uniform Resource Locator (URL) comparator 350 coupling the first comparator 320 to the logical engine 3330.** A record-keeping engine 335 is coupled between the logical engine 333 and the event log 245.

> **The security program 255 operates in conjunction with the security database 240, which includes security policies 305, known Downloadables 307, known Certificates**

> **309 and Downloadable Security Profile (DSP) data 310 corresponding to the known Downloadables 307.** Security policies 305 includes policies specific to particular users 260 and default (or generic) policies for determining whether to allow or block an incoming Downloadable. These security policies 305 may identify specific Downloadables to block, specific Downloadables to allow, or necessary criteria for allowing an unknown Downloadable. Referring to FIG. 4, security policies 305 include policy selectors 405, access control lists 410, trusted certificate lists 415, URL rule bases 420, and lists 425 of Downloadables to allow or to block per administrative override." Col. 2, line 42 – col. 3, line 28.

As is clear from the excerpts, the Touboul reference does not teach execution as part of the analysis of the security program 255. Instead, the security program uses comparators to compare whether the code matches previously trusted code. Any reference to "runtime" refers to the execution of the security program 255 (and not to the downloadable).

Second, the claims recite that the steps of first determining whether the target code includes restricted code, and then preventing the execution of the target code if it is determined that the target code includes restricted code. See claim 6 ("a determination unit configured to determine, when the application program is executed, whether a target code is included in the restriction code" and "a prevention unit configured to prevent the execution unit from executing the target code if the determination unit determines that the target code is included in the restriction code in order to prevent the target code from accessing at least a part of the data whose protection flag is indicative of receiving security protection"); see also claim 19. Thus, even if one assumed that Touboul reference teaches executing the "downloadable" to determine whether it contains a virus (which applicants do not), it would still not teach the claims as recited.

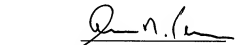
Third, the claims presently recite that "the stored data requires security protection and wherein the protection flag indicates that the stored data requires security protection". See claims 6 and 19. If the Touboul reference discloses a memory that stores data requiring security protection, it clearly conflicts with the arguments in the Office Action. Specifically, if, as stated in the Office Action, the Internal Network Security System 110 in the Touboul reference executes instruction codes to determine whether the codes include a restricted code, stored data will be accessed by the restricted code, even if the stored data require the security protection. Thus, the Internal Network Security System 110 becomes useless if the Internal Network Security System 110 in Touboul executes instruction codes to determine whether the codes include a restricted code. Clearly, this cannot be the case. The Internal Network Security System 110 cannot execute the "downloadable" because the Internal Network Security System

110 would quickly be overrun with viruses and access the data that requires protection. Therefore, Applicants respectfully contend that the Touboul reference (either alone or in combination with the other cited references) does not teach or suggest the invention as presently claimed.

SUMMARY

Applicants submit that based on the foregoing remarks, the rejections have been traversed, and that the claims are in condition for allowance. Should there be any remaining formalities, the Examiner is invited to contact the undersigned attorneys for the Applicants via telephone if such communication would expedite this application.

Respectfully submitted,



Amir N. Penn
Registration No. 40,767
Attorney for Applicant

BRINKS HOFER GILSON & LIONE
P.O. BOX 10395
CHICAGO, ILLINOIS 60610
(312) 321-4200